

The Cover Time of Deterministic Random Walks

Tobias Friedrich

Max-Planck-Institut für Informatik
Campus E1.4, 66123 Saarbrücken
Germany

Thomas Sauerwald

Simon Fraser University
Burnaby B.C. V5A 1S6
Canada

Abstract

The rotor router model is a popular deterministic analogue of a random walk on a graph. Instead of moving to a random neighbor, the neighbors are served in a fixed order. We examine how fast this “deterministic random walk” covers all vertices (or all edges). We present general techniques to derive upper bounds for the vertex and edge cover time and derive matching lower bounds for several important graph classes. Depending on the topology, the deterministic random walk can be asymptotically faster, slower or equally fast as the classic random walk. We also examine the short term behavior of deterministic random walks, that is, the time to visit a fixed small number of vertices or edges.

1 Introduction

We examine the cover time of a simple deterministic process known under various names such as “rotor router model” or “Propp machine.” It can be viewed as an attempt to derandomize random walks on graphs $G = (V, E)$. In the model each vertex $x \in V$ is equipped with a “rotor” together with a fixed sequence of the neighbors of x called “rotor sequence.” While a particle (chip, coin, . . .) performing a random walk leaves a vertex in a random direction, the deterministic random walk always goes in the direction the rotor is pointing. After a particle is sent, the rotor is updated to the next position of its rotor sequence. We examine how fast this model covers all vertices and/or edges, when one particle starts a walk from an arbitrary vertex.

1.1 Deterministic random walks

The idea of rotor routing appeared independently several times in the literature. First under the name “Eulerian walker” by Priezzhev et al. [46], then by Wagner, Lindenbaum, and Bruckstein [51] as “edge ant walk” and later by Dumitriu, Tetali, and Winkler [29] as “whirling tour.” Around the same time it was also popularized by James Propp [38] and analyzed by Cooper and Spencer [20] who called it the “Propp machine.” Later the term “deterministic random walk” was established in Doerr et al. [21, 25]. For brevity, we omit the “random” and just refer to “deterministic walk.”

Cooper and Spencer [20] showed the following remarkable similarity between the expectation of a random walk and a deterministic walk with cyclic rotor sequences: If an (almost) arbitrary distribution of particles is placed on the vertices of an infinite grid \mathbb{Z}^d and does a simultaneous walk in the deterministic walk model, then at all times and on each vertex, the

Graph class G	Vertex cover time $\mathbf{VC}(G)$ of the random walk	Vertex cover time $\widetilde{\mathbf{VC}}(G)$ of the deterministic walk
k -ary tree, $k = \mathcal{O}(1)$	$\Theta(n \log^2 n)$ [55, Cor. 9]	$\Theta(n \log n)$ (Thm. 4.2 and 3.17)
star	$\Theta(n \log n)$ [55, Cor. 9]	$\Theta(n)$ (Thm. 4.1)
cycle	$\Theta(n^2)$ [43, Ex. 1]	$\Theta(n^2)$ (Thm. 4.3 and 3.15)
lollipop graph	$\Theta(n^3)$ [43, Thm. 2.1]	$\Theta(n^3)$ (Thm. 4.4 and 3.18)
expander	$\Theta(n \log n)$ [11, Cor. 6], [49]	$\Theta(n \log n)$ (Thm. 4.5, Cor. 3.11)
two-dim. torus	$\Theta(n \log^2 n)$ [55, Thm. 4], [13, Thm. 6.1]	$\Theta(n^{1.5})$ (Thm. 4.7 and 3.15)
d -dim. torus ($d \geq 3$)	$\Theta(n \log n)$ [55, Cor. 12], [13, Thm. 6.1]	$\mathcal{O}(n^{1+1/d})$ (Thm. 3.15)
hypercube	$\Theta(n \log n)$ [1, p. 372], [45, Sec. 5.2]	$\Theta(n \log^2 n)$ (Thm. 4.8 and 3.16)
complete	$\Theta(n \log n)$ [43, Ex. 1]	$\Theta(n^2)$ (Thm. 4.1 and 3.14)

Table 1: Comparison of the vertex cover time of random and deterministic walk on different graphs ($n = |V|$).

number of particles deviates from the expected number the standard random walk would have gotten there, by at most a constant. This constant is precisely known for the cases $d = 1$ [21] and $d = 2$ [25]. It is further known that there is no such constant for infinite trees [22]. Levine and Peres [42] also extensively studied a related model called internal diffusion-limited aggregation [40, 41] for deterministic walks.

As in these works, our aim is to understand random walk and their deterministic counterpart from a theoretical viewpoint. However, we would like to mention that the rotor router mechanism also led to improvements in applications. With a random initial rotor direction, the quasirandom rumor spreading protocol broadcasts faster in some networks than its random counterpart [4, 26–28]. A similar idea is used in quasirandom external mergesort [9] and quasirandom load balancing [33].

We consider our model of a deterministic walk based on rotor routing to be a simple and canonic derandomization of a random walk which is not tailored for search problems. On the other hand, there is a vast literature on local deterministic agents/robots/ants patrolling or covering all vertices or edges of a graph (e.g. [34, 39, 48, 50, 51]). For instance, Cooper, Ilcinkas, Klasing, and Kosowski [19] studied a model where the walk uses adjacent edges which have been traversed the smallest number of times. However, all of these models are more specialized and require additional counters/identifiers/markers/pebbles on the vertices or edges of the explored graph.

1.2 Cover time of random walks

In his survey, Lovász [43] mentions three important measures of a random walk: cover time, hitting time, and mixing time. These three (especially the first two) are closely related, here we will mainly concentrate on the *cover time* which is the expected number of steps to visit every node. The study of the cover time of random walks on graphs was initiated in 1979. Motivated by the space-complexity of the s - t -connectivity problem, Aleliunas et al. [3] showed that the cover time is upper bounded by $\mathcal{O}(|V| |E|)$ for any graph. For regular graphs, Feige [31] gave an improved upper bound of $\mathcal{O}(|V|^2)$ for the cover time. Broder and Karlin [11] proved several bounds which rely on the spectral gap of the transition matrix. Their bounds imply that the cover time on a regular expander graph is $\Theta(|V| \log |V|)$. In addition, many papers are devoted to the study of the cover time on special graphs such as hypercubes [1], random graphs [15–17], random regular graphs [14], random geometric

Graph class G	Edge cover time $\text{EC}(G)$ of the random walk		Edge cover time $\widetilde{\text{EC}}(G)$ of the deterministic walk	
k -ary tree, $k = \mathcal{O}(1)$	$\Theta(n \log^2 n)$	[55, Cor. 9]	$\Theta(n \log n)$	(Thm. 4.2 and 3.17)
star	$\Theta(n \log n)$	[55, Cor. 9]	$\Theta(n)$	(Thm. 4.1)
complete	$\Theta(n^2 \log n)$	[54, 55]	$\Theta(n^2)$	(Thm. 4.1 and 3.14)
expander	$\Theta(n \log n)$	[54, 55]	$\Theta(n \log n)$	(Thm. 4.5, Cor. 3.11)
cycle	$\Theta(n^2)$	[43, Ex. 1]	$\Theta(n^2)$	(Thm. 4.3 and 3.15)
lollipop graph	$\Theta(n^3)$	[43, Thm. 2.1], [54, Lem. 2]	$\Theta(n^3)$	(Thm. 4.4 and 3.18)
hypercube	$\Theta(n \log^2 n)$	[54, 55]	$\Theta(n \log^2 n)$	(Thm. 4.8 and 3.16)
two-dim. torus	$\Theta(n \log^2 n)$	[54, 55]	$\Theta(n^{1.5})$	(Thm. 4.7 and 3.15)
d -dim. torus ($d \geq 3$)	$\Theta(n \log n)$	[54, 55]	$\mathcal{O}(n^{1+1/d})$	(Thm. 3.15)

Table 2: Comparison of the edge cover time of random and deterministic walk on different graphs ($n = |V|$).

graphs [18], and planar graphs [37]. A general lower bound of $(1 - o(1))|V| \ln |V|$ for any graph was shown by Feige [30].

A natural variant of the cover time is the so-called *edge cover time*, which measures the expected number of steps to traverse all edges. Amongst other results, Zuckerman [54, 55] proved that the edge cover time of general graphs is at least $\Omega(|E| \log |E|)$ and at most $\mathcal{O}(|V||E|)$. Finally, Barnes and Feige [7, 8] considered the time until a certain number of vertices (or edges) has been visited.

1.3 Cover time of deterministic walks (our results)

For the case of a cyclic rotor sequence the edge cover time is known to be $\Theta(|E| \text{diam}(G))$ (see Yanovski et al. [53] for the upper and Bampas et al. [6] for the lower bound). It is further known that there are rotor sequences such that the edge cover time is precisely $|E|$ [46]. We allow *arbitrary* rotor sequences and present three techniques to upper bound the edge cover time based on the local divergence (Thm. 3.5), expansion of the graph (Thm. 3.10), and a corresponding flow problem (Thm. 3.13). With these general theorems it is easy to prove upper bounds for expanders, complete graphs, torus graphs, hypercubes, k -ary trees and lollipop graphs. Though these bounds are known to be tight, it is illuminating to study which setup of the rotors matches these upper bounds. This is the motivation for Section 4 which presents matching lower bounds for all forementioned graphs by describing the precise setup of the rotors.

It is not our aim to prove superiority of the deterministic walk, but it is instructive to compare our results for the vertex and edge cover time with the respective bounds of the random walk. Tables 1 and 2 group the graphs in three classes depending whether random or deterministic walk is faster. In spite of the strong adversary (as the order of the rotors is completely arbitrary), the deterministic walk is surprisingly efficient. It is known that the edge cover time of random walks can be asymptotically larger than its vertex cover time. Somewhat unexpectedly, this is not the case for the deterministic walk. To highlight this issue, let us consider hypercubes and complete graph. For these graphs, the vertex cover time of the deterministic walk is larger while the edge cover time is smaller (complete graph) or equal (hypercube) compared to the random walk.

Analogous to the results of Barnes and Feige [7, 8] for random walks, we also analyze the short term behavior of the deterministic walk in Section 5. As an example observe that

Theorem 5.1 proves that for $1 \leq \alpha < 2$ the deterministic walk only needs $\mathcal{O}(|V|^\alpha)$ steps to visit $|V|^\alpha$ edges of any graph with minimum degree $\Omega(n)$ while the random walk needs $\mathcal{O}(|V|^{2\alpha-1})$ steps according to [7, 8] (cf. Table 4).

2 Models and Preliminaries

2.1 Random Walks

We consider weighted random walks on finite connected graphs $G = (V, E)$. For this, we assign every pair of vertices $u, v \in V$ a weight $c(u, v) \in \mathbb{N}_0$ (rational weights can be handled by scaling) such that $c(u, v) = c(v, u) > 0$ if $\{u, v\} \in E$ and $c(u, v) = c(v, u) = 0$ otherwise. This defines transition probabilities $\mathbf{P}_{u,v} := c(u, v)/c(u)$ with $c(u) := \sum_{w \in V} c(u, w)$. So, whenever a random walk is at a vertex u it moves to a vertex v in the next step with probability $\mathbf{P}_{u,v}$. Moreover, note that for all $u, v \in V$, $c(u, v) = c(v, u)$ while $\mathbf{P}_{u,v} \neq \mathbf{P}_{v,u}$ in general. This defines a time-reversible, irreducible, finite Markov chain X_0, X_1, \dots with transition matrix \mathbf{P} (cf. [2]). The t -step probabilities of the walk can be obtained by taking the t -th power of \mathbf{P}^t . In what follows, we prefer to use the term weighted random walk instead of Markov chain to emphasize the limitation to rational transition probabilities.

It is intuitively clear that a random walk with large weights $c(u, v)$ is harder to approximate deterministically with a simple rotor sequence. To measure this, we use $c_{\max} := \max_{u,v \in V} c(u, v)$. An important special case is the *unweighted random walk* with $c(u, v) \in \{0, 1\}$ for all $u, v \in V$ on a simple graph. In this case, $\mathbf{P}_{u,v} = 1/\deg(u)$ for all $\{u, v\} \in E$, and $c_{\max} = 1$. Our general results hold for weighted (random) walks. However, the derived bounds for specific graphs are only stated for unweighted walks. With *random walk* we mean unweighted random walk and if a random walk is allowed to be weighted we will emphasize this.

For weighted and unweighted random walks we define for a graph G ,

- cover time: $\text{VC}(G) = \max_{u \in V} \mathbf{E} \left[\min \{t \geq 0: \bigcup_{\ell=0}^t \{X_\ell\} = V\} \mid X_0 = u \right]$,
- edge cover time: $\text{EC}(G) = \max_{u \in V} \mathbf{E} \left[\min \{t \geq 0: \bigcup_{\ell=1}^t \{X_{\ell-1}, X_\ell\} = E\} \mid X_0 = u \right]$.

The (edge) cover time of a graph class \mathcal{G} is the maximum of the (edge) cover times of all graphs of the graph class. Observe that $\text{VC}(\mathcal{G}) \leq \text{EC}(\mathcal{G})$ for all graphs G . For vertices $u, v \in V$ we further define

- (expected) hitting time: $\text{H}(u, v) = \mathbf{E} [\min \{t \geq 0: X_t = v\} \mid X_0 = u]$,
- stationary distribution: $\pi_u = c(u) / \sum_{w \in V} c(w)$.

2.2 Deterministic Random Walks

We define weighted deterministic random walks (or short: weighted deterministic walks) based on rotor routers as introduced by Holroyd and Propp [35]. For a weighted random walk, we define the corresponding weighted deterministic walk as follows. We use a tilde (\sim) to mark variables related to the deterministic walk. To each vertex u we assign a rotor sequence $\tilde{s}(u) = (\tilde{s}(u, 1), \tilde{s}(u, 2), \dots, \tilde{s}(u, \tilde{d}(u))) \in V^{\tilde{d}(u)}$ of arbitrary length $\tilde{d}(u)$ such that the number of times a neighbor v occurs in the rotor sequence $\tilde{s}(u)$ corresponds to the transition probability to go from u to v in the weighted random walk, that is, $\mathbf{P}_{u,v} = |\{i \in [\tilde{d}(u)]: \tilde{s}(u, i) = v\}| / \tilde{d}(u)$ with $[\tilde{d}(u)] := \{1, \dots, \tilde{d}(u)\}$. For a weighted random walk, $\tilde{d}(u)$

is a multiple of the lowest common denominator of the transition probabilities from u to its neighbors. For the standard random walk, a corresponding canonical deterministic walk would be $\tilde{d}(u) = \deg(u)$ and a permutation of the neighbors of u as rotor sequence $\tilde{s}(u)$. As the length of the rotor sequences crucially influences the performance of a deterministic walk, we set $\tilde{\kappa} := \max_{u \in V} \tilde{d}(u) / \deg(u)$ (note that $\tilde{\kappa} \geq 1$). The set V together with $\tilde{s}(u)$ and $\tilde{d}(u)$ for all $u \in V$ defines the deterministic walk, sometimes abbreviated **D**. Note that every deterministic walk has a unique corresponding random walk while there are many deterministic walks corresponding to one random walk.

We also assign to each vertex u an integer $\tilde{r}_t(u) \in [\tilde{d}(u)]$ corresponding to a rotor at u pointing to $\tilde{s}(u, \tilde{r}_t(u))$ at step t . A *rotor configuration* C describes the rotor sequences $\tilde{s}(u)$ and initial rotor directions $\tilde{r}_0(u)$ for all vertices $u \in V$. At every time step t the walk moves from \tilde{x}_t in the direction of the current rotor of \tilde{x}_t and this rotor is incremented¹ to the next position according to the rotor sequence $\tilde{s}(\tilde{x}_t)$ of \tilde{x}_t . More formally, for given \tilde{x}_t and $\tilde{r}_t(\cdot)$ at time $t \geq 0$ we set $\tilde{x}_{t+1} := s(\tilde{x}_t, \tilde{r}_t(\tilde{x}_t))$, $\tilde{r}_{t+1}(\tilde{x}_t) := \tilde{r}_t(\tilde{x}_t) \bmod \tilde{d}(\tilde{x}_t) + 1$, and $\tilde{r}_{t+1}(u) := \tilde{r}_t(u)$ for all $u \neq \tilde{x}_t$. Let \mathcal{C} be the set of all possible rotor configurations (that is, $\tilde{s}(u)$, $\tilde{r}_0(u)$ for $u \in V$) of a corresponding deterministic walk for a fixed weighted random walk (and fixed rotor sequence length $\tilde{d}(u)$ for each $u \in V$). Given a rotor configuration $C \in \mathcal{C}$ and an initial location $\tilde{x}_0 \in V$, the vertices $\tilde{x}_0, \tilde{x}_1, \dots \in V$ visited by a deterministic walk are completely determined.

For deterministic walks we define for a graph G and vertices $u, v \in V$,

- deterministic cover time: $\widetilde{\text{VC}}(G) = \max_{\tilde{x}_0 \in V} \max_{C \in \mathcal{C}} \min \{t \geq 0: \bigcup_{\ell=0}^t \{\tilde{x}_\ell\} = V\}$,
- deterministic edge cover time:
 $\widetilde{\text{EC}}(G) = \max_{\tilde{x}_0 \in V} \max_{C \in \mathcal{C}} \min \{t \geq 0: \bigcup_{\ell=1}^t \{\tilde{x}_{\ell-1}, \tilde{x}_\ell\} = E\}$,
- hitting time: $\widetilde{\text{H}}(u, v) = \max_{C \in \mathcal{C}} \min \{t \geq 0: \tilde{x}_t = u, \tilde{x}_0 = v\}$.

Note that the definition of the deterministic cover time takes the *maximum* over all possible rotor configurations, while the cover time of a random walk takes the *expectation* over the random decisions. Also, $\widetilde{\text{VC}}(G) \leq \widetilde{\text{EC}}(G)$ for all graphs G . We further define for fixed configurations $C \in \mathcal{C}$, \tilde{x}_0 , and vertices $u, v \in V$,

- number of visits to vertex u : $\tilde{N}_t(u) = |\{0 \leq \ell \leq t: \tilde{x}_\ell = u\}|$,
- number of traversals of a directed edge $u \rightarrow v$:
 $\tilde{N}_t(u \rightarrow v) = |\{1 \leq \ell \leq t: (\tilde{x}_{\ell-1}, \tilde{x}_\ell) = (u, v)\}|$.

2.3 Graph-Theoretic Notation

We consider finite, connected graphs $G = (V, E)$. Unless stated differently, $n := |V|$ is the number vertices and $m := |E|$ the number of undirected edges. By δ and Δ we denote the minimum and maximum degree of the graph, respectively. For a pair of vertices $u, v \in V$, we denote by $\text{dist}(u, v)$ their distance, i.e., the length of a shortest path between them. For a vertex $u \in V$, let $\Gamma(u)$ denote the set of all neighbors of u . More generally, for any $k \geq 1$, $\Gamma^k(u)$ denotes the set of vertices v with $\text{dist}(u, v) = k$. For any subsets $S, T \subseteq V$, $E(S)$ denotes the set of edges with one endpoint in S and $E(S, T)$ denotes the edges $\{u, v\}$ with $u \in S$ and $v \in T$. As a walk is something directed, we also have to argue about directed edges though our graph G is undirected. In slight abuse of notation, for $\{u, v\} \in E$ we might also write $(u, v) \in E$ or $(v, u) \in E$. Finally, all logarithms used here are to the base of 2.

¹In this respect we slightly deviate from the model of Holroyd and Propp [35] who first increment the rotor and then move the chip, but this change is insignificant here.

3 Upper Bounds on the Deterministic Cover Times

Very recently, Holroyd and Propp [35] proved that several natural quantities of the weighted deterministic walk as defined in Section 2.2 concentrate around the respective expected values of the corresponding weighted random walk. To state their result formally, we set for a vertex $v \in V$,

$$K(v) := \max_{u \in V} H(u, v) + \frac{1}{2} \left(\frac{\tilde{d}(v)}{\pi_v} + \sum_{i, j \in V} \tilde{d}(i) \mathbf{P}_{i, j} |H(i, v) - H(j, v) - 1| \right). \quad (1)$$

Theorem 3.1 ([35, Thm. 4]). *For all weighted deterministic walks, all vertices $v \in V$, and all times t ,*

$$\left| \pi_v - \frac{\tilde{N}_t(v)}{t} \right| \leq \frac{K(v) \pi_v}{t}.$$

Roughly speaking, Theorem 3.1 states that the proportion of time spent by the weighted deterministic walk concentrates around the stationary distribution for all configurations $C \in \mathcal{C}$ and all starting points \tilde{x}_0 . To quantify the hitting or cover time with Theorem 3.1, we choose $t = K(v) + 1$ to get $\tilde{N}_t(v) > 0$. To get a bound for the edge cover time, we choose $t = 3K(v)$ and observe that then $\tilde{N}_t(v) \geq 2\pi_v K(v) > \tilde{d}(v)$. This already shows the following corollary.

Corollary 3.2. *For all weighted deterministic walks,*

$$\begin{aligned} \tilde{H}(u, v) &\leq K(v) + 1 && \text{for all } u, v \in V, \\ \widetilde{VC}(G) &\leq \max_{v \in V} K(v) + 1, \\ \widetilde{EC}(G) &\leq 3 \max_{v \in V} K(v). \end{aligned}$$

One obvious question that arises from Theorem 3.1 and Corollary 3.2 is how to bound the value $K(v)$. While it is clear that $K(v)$ is polynomial in n (provided that c_{\max} and $\tilde{\kappa}$ are polynomially bounded), it is not clear how to get more precise upper bounds. A key tool to tackle the difference of hitting times in $K(v)$ is the following elementary lemma, where in case of a periodic walk the sum is taken as a Cesáro summation [12].

Lemma 3.3. *For all weighted random walks and all vertices $i, j, v \in V$,*

$$\sum_{t=0}^{\infty} (\mathbf{P}_{i, v}^t - \mathbf{P}_{j, v}^t) = \pi_v (H(j, v) - H(i, v)).$$

Proof. Let \mathbf{Z} be the fundamental matrix of \mathbf{P} defined as $\mathbf{Z}_{ij} := \sum_{t=0}^{\infty} (\mathbf{P}_{i, j}^t - \pi_j)$. It is known that for any pair of vertices i and v , $\pi_v H(i, v) = Z_{vv} - Z_{iv}$ (cf. [2, Ch. 2, Lem. 12]). Hence by the convergence of \mathbf{P} ,

$$\begin{aligned} \pi_v (H(j, v) - H(i, v)) &= (Z_{vv} - Z_{jv}) - (Z_{vv} - Z_{iv}) \\ &= \sum_{t=0}^{\infty} (\mathbf{P}_{i, v}^t - \pi_v) - \sum_{t=0}^{\infty} (\mathbf{P}_{j, v}^t - \pi_v) = \sum_{t=0}^{\infty} (\mathbf{P}_{i, v}^t - \mathbf{P}_{j, v}^t). \quad \square \end{aligned}$$

3.1 Bounding $K(v)$ by the local divergence

To analyze weighted random walks, we use the notion of local divergence which has been a fundamental quantity in the analysis of load balancing algorithms [32, 47]. Moreover, the local divergence is considered to be of independent interest (see [47] and further references therein).

Definition 3.4. *The local divergence of a weighted random walk is $\Psi(\mathbf{P}) := \max_{v \in V} \Psi(\mathbf{P}, v)$, where $\Psi(\mathbf{P}, v)$ is the local divergence w.r.t. to a vertex $v \in V$ defined as $\Psi(\mathbf{P}, v) := \sum_{t=0}^{\infty} \sum_{\{i,j\} \in E} |\mathbf{P}_{i,v}^t - \mathbf{P}_{j,v}^t|$.*

Using Corollary 3.2 and Lemma 3.3, we get the following bound on the hitting time of a deterministic walk.

Theorem 3.5. *For all deterministic walks and all vertices $v \in V$,*

$$K(v) \leq \max_{u \in V} H(u, v) + \frac{\tilde{\kappa} c_{\max}}{\pi_v} \Psi(\mathbf{P}, v) + 2m \tilde{\kappa} c_{\max}.$$

Proof. To bound $K(v)$ we first observe that by definition of $\tilde{\kappa}$ and c_{\max} for all $u, v \in V$,

$$\frac{\tilde{d}(v)}{\pi_v} = \frac{\tilde{d}(v) \sum_{i,j \in V} c(i, j)}{c(v)} \leq \frac{\tilde{\kappa} \deg(v) 2 \sum_{\{i,j\} \in E} c(i, j)}{c(v)} \leq 2\tilde{\kappa} \sum_{\{i,j\} \in E} c(i, j) \leq 2m \tilde{\kappa} c_{\max},$$

$$\tilde{d}(u) \mathbf{P}_{u,v} \leq \tilde{\kappa} \deg(u) \mathbf{P}_{u,v} = \frac{\tilde{\kappa} \deg(u) c(u, v)}{c(u)} \leq \tilde{\kappa} c(u, v) \leq \tilde{\kappa} c_{\max}.$$

Therefore,

$$\begin{aligned} K(v) &\leq \max_{u \in V} H(u, v) + m \tilde{\kappa} c_{\max} + \frac{1}{2} \sum_{i,j \in V} \tilde{\kappa} c_{\max} (|H(i, v) - H(j, v)| + 1) \\ &\leq \max_{u \in V} H(u, v) + 2m \tilde{\kappa} c_{\max} + \tilde{\kappa} c_{\max} \sum_{\{i,j\} \in E} |H(i, v) - H(j, v)| \\ &\leq \max_{u \in V} H(u, v) + 2m \tilde{\kappa} c_{\max} + \frac{\tilde{\kappa} c_{\max}}{\pi_v} \Psi(\mathbf{P}, v), \end{aligned}$$

where the last inequality follows from Lemma 3.3 and Definition 3.4. \square

To see where the dependence on $\tilde{\kappa}$ in Theorem 3.5 comes from, remember that our bounds hold for all configurations $C \in \mathcal{C}$ of the deterministic walk. This is equivalent to bounds for a walk where an adversary chooses the rotor sequences within the given setting. Hence a larger $\tilde{\kappa}$ strengthens the adversary as it gets more freedom of choice in the order of the rotor sequence. On the other hand, the c_{\max} measures how skewed the probability distribution of the random walk can be. With larger c_{\max} , they get harder to approximate deterministically.

Note that Theorem 3.5 is more general than just giving an upper bound for hitting and cover times via Corollary 3.2. It can be useful in the other directions, too. To give a specific example, we can apply the result of Theorem 4.8 that $\widetilde{\text{EC}}(G) = \Omega(n \log^2 n)$ for hypercubes and $\max_{u,v} H(u, v) = \mathcal{O}(n)$ (cf. [43]) to Theorem 3.5 and obtain a lower bound of $\Omega(n \log^2 n)$ on the local divergence of hypercubes.

3.2 Bounding $K(v)$ for symmetric walks

To get meaningful bounds for the cover time, we restrict to unweighted random walks in the following. In our notation this implies $c_{\max} = 1$ while $\tilde{\kappa}$ is still arbitrary. First, we derive a tighter version of Theorem 3.5 for symmetric random walks defined as follows.

Definition 3.6. A symmetric random walk has transition probabilities $\mathbf{P}'_{u,v} = \frac{1}{\Delta+1}$ if $\{u, v\} \in E$, $\mathbf{P}'_{u,u} = 1 - \frac{1}{\Delta+1} \deg(u)$ and $\mathbf{P}'_{u,v} = 0$ otherwise.

These symmetric random walks occur frequently in the literature, e.g., for load balancing [32, 47] or for the cover time [5]. The corresponding deterministic walk is defined as follows.

Definition 3.7. For an unweighted deterministic walk \mathbf{D} with rotor sequences $\tilde{s}(\cdot)$ of length $\tilde{d}(\cdot)$, let the corresponding symmetric deterministic walk \mathbf{D}' have for all $u \in V$ rotor sequences $\tilde{s}'(u)$ of length $\tilde{d}'(u) := \frac{\Delta+1}{\deg(u)} \tilde{d}(u)$. with $\tilde{s}'(u, i) := \tilde{s}(u, i)$ for $i \leq \tilde{d}(u)$ and $\tilde{s}'(u, i) := u$ for $i > \tilde{d}(u)$.

It is easy to verify that the definition “commutes”, that is, for a deterministic walk \mathbf{D} corresponding to a random walk \mathbf{P} , the corresponding deterministic walk \mathbf{D}' corresponds to the corresponding symmetric random walk \mathbf{P}' . Let

$$\begin{array}{ccc} \mathbf{P} & \longrightarrow & \mathbf{D} \\ & \downarrow & \downarrow \\ \mathbf{P}' & \longrightarrow & \mathbf{D}' \end{array}$$

all primed variables (π'_u , $K'(v)$, κ' , $c'(u, v)$, c'_{\max} , $H'(u, v)$, $\tilde{H}'(u, v)$, $\mathbf{VC}'(G)$, $\mathbf{WC}'(G)$, $\mathbf{EC}'(G)$, $\mathbf{EC}'(G)$) have their natural meaning for the symmetric random walk and symmetric deterministic walk.

As \mathbf{P}' is symmetric, the stationary distribution of \mathbf{P}' is uniform, i.e., $\pi'_i = 1/n$ for all $i \in V$. Note that the symmetric walk is in fact a weighted walk with $c'(u, v) = 1$ for $\{u, v\} \in E$, $c'(u, u) = \Delta + 1 - \deg(u)$ for $u \in V$, and $c'(u, v) = 0$ otherwise. Using $c'_{\max} = \Delta + 1 - \delta$ in Theorem 3.5 is too coarse. To get a better bound on $K'(v)$ for symmetric walks, observe that for all $v \in V$

$$\frac{\tilde{d}'(v)}{\pi'(v)} = n \tilde{d}(v) \frac{\Delta + 1}{\deg(v)} \leq n \tilde{\kappa} (\Delta + 1) \quad (2)$$

and for all $\{u, v\} \in E$

$$\tilde{d}'(u) \mathbf{P}'_{u,v} = \frac{\tilde{d}(u)}{\deg(u)} \leq \tilde{\kappa}. \quad (3)$$

Plugging this in the definition of $K(v)$ as in Theorem 3.5 gives the following theorem.

Theorem 3.8. For all symmetric deterministic walks and all vertices $v \in V$,

$$K'(v) = \mathcal{O} \left(\max_{u \in V} H'(u, v) + \frac{\tilde{\kappa}}{\pi'(v)} \Psi(\mathbf{P}', v) + n \Delta \tilde{\kappa} \right).$$

By definition, $\mathbf{EC}(G) \leq \mathbf{EC}'(G)$ and $H(u, v) \leq H'(u, v)$ for all $u, v \in V$. The following lemma gives a natural reverse of the latter inequality.

Lemma 3.9. For a random walk \mathbf{P} and a symmetric random walk \mathbf{P}' it holds for any pair of vertices u, v that

$$H'(u, v) \leq \frac{\Delta + 1}{\delta} H(u, v).$$

Proof. Let us consider the transition matrix \mathbf{P}'' with $\mathbf{P}''_{u,u} = 1 - \frac{\delta}{\Delta+1}$, $\mathbf{P}''_{u,v} = \frac{\delta}{\Delta+1} \cdot \frac{1}{\deg(u)}$ if $\{u, v\} \in E$ and $\mathbf{P}''_{v,v} = 0$ otherwise. Let H'' denote the hitting times of a random walk according to \mathbf{P}'' . We couple the non-loop steps of a random walk according to \mathbf{P}' with the non-loop steps of a random walk according to \mathbf{P}'' , as in both walks, a neighbor is chosen uniformly at random (conditioned on the event that the walk does not loop).

Since all respective loop-probabilities satisfy $\mathbf{P}'_{u,u} \leq \mathbf{P}''_{u,u}$, it follows that for all vertices $u, v \in V$, $H'(u, v) \leq H''(u, v)$. Our next aim is to relate $\tau''(u, v)$ to $\tau(u, v)$, where τ'' (τ , resp.) is the first step when a random walk according to \mathbf{P}'' (\mathbf{P} , resp.) starting at u visits v . We can again couple the non-loop steps of both random walks, since every non-loop step of \mathbf{P}'' chooses a uniform neighbor and so does \mathbf{P} . Hence, $H''(u, v) = \mathbf{E}[\tau''(u, v)] = \mathbf{E}[\sum_{i=1}^{\tau''(u, v)} X_i]$, where the X_i 's are independent, identically distributed geometric random variable with mean $\frac{\Delta+1}{\delta}$. Applying Wald's equation [52] yields

$$H''(u, v) = \mathbf{E}[\tau(u, v)] \cdot \mathbf{E}[X_1] = H(u, v) \cdot \frac{\Delta+1}{\delta},$$

which proves the claim. \square

3.3 Upper bound on the deterministic cover time depending on the expansion

We now derive an upper bound for $\widetilde{\text{EC}}(G)$ that depends on the expansion properties of G . Let $\lambda_2(\mathbf{P})$ be the second-largest eigenvalue in absolute value of \mathbf{P} .

Theorem 3.10. *For all graphs G , $\widetilde{\text{EC}}(G) = \mathcal{O}(\frac{\Delta}{\delta} \frac{n}{1-\lambda_2(\mathbf{P})} + n \tilde{\kappa} \frac{\Delta}{\delta} \frac{\Delta \log n}{1-\lambda_2(\mathbf{P})})$.*

Proof. Let \mathbf{P} and \mathbf{D} be corresponding unweighted random and deterministic walks and \mathbf{P}' and \mathbf{D}' be defined as in Definitions 3.6 and 3.7. From the latter definition we get $\widetilde{\text{EC}}(G) \leq \widetilde{\text{EC}}'(G)$, as additional loops in the rotor sequence can only slow down the covering process. Hence it suffices to bound $\widetilde{\text{EC}}'(G)$ with Theorem 3.8. We will now upper bound all three summands involved in Theorem 3.8.

By two classical result for reversible, ergodic Markov chains ([2, Chap. 3, Lem. 15] and [2, Chap. 3, Lem. 17] of Aldous and Fill),

$$\max_{u,v} H'(u, v) \leq 2 \sum_{u \in V} \pi_u \cdot H'(u, v) \leq 2 \frac{1 - \pi_v}{\pi_v \cdot (1 - \lambda_2(\mathbf{P}'))}.$$

As \mathbf{P}' is symmetric, the stationary distribution of \mathbf{P}' is uniform and therefore

$$\max_{u,v \in V} H'(u, v) \leq 2 \frac{n}{1 - \lambda_2(\mathbf{P}')}.$$

In order to relate $\lambda_2(\mathbf{P})$ and $\lambda_2(\mathbf{P}')$, we use the following “direct comparison lemma” for reversible Markov Chains \mathbf{P} and \mathbf{P}' from [23, Eq. 2.3] (where in their notation, we plug in $a = \min_{i \in V} \frac{\pi_i}{\pi'_i}$ and $A = \max_{(i,j) \in E, i \neq j} \frac{\pi_i \mathbf{P}_{i,j}}{\pi'_j \mathbf{P}'_{i,j}}$) to obtain that

$$\frac{1 - \lambda_2(\mathbf{P})}{1 - \lambda_2(\mathbf{P}')} \leq \frac{\max_{(i,j) \in E, i \neq j} \frac{\pi_i \mathbf{P}_{i,j}}{\pi'_i \mathbf{P}'_{i,j}}}{\min_{i \in V} \frac{\pi_i}{\pi'_i}}.$$

We now determine the denominator and numerator of the right hand side of equation (5). As $\pi'_i = 1/n$ and $\pi_i = \frac{\deg(i)}{2m}$ for all $i \in V$, $\min_i \frac{\pi_i}{\pi'_i} = \frac{\delta}{2m}n$. Moreover, for any edge $\{i, j\} \in E$, $\pi_i \mathbf{P}_{i,j} = \frac{\deg(i)}{2m} \frac{1}{\deg(i)} = \frac{1}{2m}$ and $\pi'_i \mathbf{P}'_{i,j} = \frac{1}{n} \frac{1}{\Delta+1}$ and therefore $\max_{(i,j) \in E, i \neq j} \frac{\pi_i \mathbf{P}_{i,j}}{\pi'_i \mathbf{P}'_{i,j}} = \frac{n(\Delta+1)}{2m}$. Plugging this in equation (5) yields

$$\frac{1 - \lambda_2(\mathbf{P})}{1 - \lambda_2(\mathbf{P}')} \leq \frac{\frac{n(\Delta+1)}{2m}}{\frac{\delta}{2m}n} = \frac{\Delta+1}{\delta}. \quad (6)$$

From Theorem 4 of Rabani et al. [47] we know the following upper bound on $\Psi(\mathbf{P}')$,

$$\Psi(\mathbf{P}') = \mathcal{O}\left(\frac{\Delta \log n}{1 - \lambda_2(\mathbf{P}')}\right). \quad (7)$$

Plugging all this in Theorem 3.8 and Corollary 3.2 gives

$$\begin{aligned} \widetilde{\text{EC}}'(G) &= \mathcal{O}\left(\max_{u \in V} H'(u, v) + \frac{\tilde{\kappa}}{\pi'(v)} \Psi(\mathbf{P}', v) + n \Delta \tilde{\kappa}\right) \\ &= \mathcal{O}\left(\frac{n}{1 - \lambda_2(\mathbf{P}')} + n \frac{\Delta \tilde{\kappa} \log n}{1 - \lambda_2(\mathbf{P}')} \right) \quad (\text{by equations (4) and (7)}) \\ &= \mathcal{O}\left(\frac{\Delta}{\delta} \frac{n}{1 - \lambda_2(\mathbf{P})} + n \frac{\Delta}{\delta} \frac{\Delta \tilde{\kappa} \log n}{1 - \lambda_2(\mathbf{P})} \right). \quad (\text{by equation (6)}) \end{aligned}$$

As $\widetilde{\text{EC}}(G) \leq \widetilde{\text{EC}}'(G)$, this finishes the proof. \square

Here, we call a graph with constant maximum degree an expander graph, if $1/(1 - \lambda_2(\mathbf{P})) = \mathcal{O}(1)$ (equivalently, we have for all subsets $X \subseteq V, 1 \leq |X| \leq n/2$, $|E(X, X^c)| = \Omega(|X|)$ (cf. [23, Prop. 6])). Using Theorem 3.10, we immediately get the following upper bound on $\widetilde{\text{EC}}(G)$ for expanders.

Corollary 3.11. *For all expander graphs, $\widetilde{\text{EC}}(G) = \mathcal{O}(\tilde{\kappa} n \log n)$.*

3.4 Upper bound on the deterministic cover time by flows

We relate the edge cover time of the unweighted random walk to the optimal solution of the following flow problem.

Definition 3.12 (cmp. [44, Def. 1, Rem. 1]). *Consider the flow problem where a distinguished source node s sends a flow amount of 1 to each other node in the graph. Then $f_s(i, j)$ denotes the load transferred along edge $\{i, j\}$ (note $f_s(i, j) = -f_s(j, i)$) such that $\sum_{\{i, j\} \in E} f_s(i, j)^2$ is minimized.*

Theorem 3.13. *For all graphs G ,*

$$\widetilde{\text{EC}}(G) = \mathcal{O}\left(\frac{\Delta}{\delta} \max_{u, v \in V} H(u, v) + \Delta n \tilde{\kappa} + \tilde{\kappa} \Delta \max_{s \in V} \sum_{\{i, j\} \in E} |f_s(i, j)|\right)$$

where f_s is the flow with source s according to Definition 3.12.

Proof. Let \mathbf{P} and \mathbf{D} be corresponding unweighted random and deterministic walks and \mathbf{P}' and \mathbf{D}' be defined as in Definitions 3.6 and 3.7.

By combining the equalities from [44, Def. 1 & Thm. 1] (where we set the flow amount sent by s to any other vertex to 1),

$$|f_s(i, j)| = \frac{n}{\Delta + 1} \left| \sum_{t=0}^{\infty} (\mathbf{P}'^t_{i,s} - \mathbf{P}'^t_{j,s}) \right| \quad \text{for any edge } \{i, j\} \in E. \quad (8)$$

Now plugging equations (2) and (3) in the definition of $K'(v)$ from equation (1) gives

$$\begin{aligned} K'(v) &= \mathcal{O} \left(\max_{u,v \in V} H'(u, v) + \Delta n \tilde{\kappa} + \tilde{\kappa} \max_{v \in V} \sum_{\{i,j\} \in E} |H'(i, v) - H'(j, v)| \right) \\ &= \mathcal{O} \left(\max_{u,v \in V} H'(u, v) + \Delta n \tilde{\kappa} + \tilde{\kappa} \max_{v \in V} n \sum_{\{i,j\} \in E} \left| \sum_{t=0}^{\infty} \mathbf{P}'^t_{iv} - \mathbf{P}'^t_{jv} \right| \right) \quad (\text{by Lemma 3.3}) \\ &= \mathcal{O} \left(\max_{u,v \in V} H'(u, v) + \Delta n \tilde{\kappa} + \tilde{\kappa} \Delta \max_{s \in V} \sum_{\{i,j\} \in E} |f_s(i, j)| \right) \quad (\text{by equation (8)}) \\ &= \mathcal{O} \left(\frac{\Delta}{\delta} \max_{u,v \in V} H(u, v) + \Delta n \tilde{\kappa} + \tilde{\kappa} \Delta \max_{s \in V} \sum_{\{i,j\} \in E} |f_s(i, j)| \right) \quad (\text{by Lemma 3.9}) \end{aligned}$$

With Corollary 3.2, $\widetilde{\text{EC}}(G) \leq \widetilde{\text{EC}}'(G) \leq 3 \max_{v \in V} K'(v)$ finishes the proof. \square

3.5 Upper bounds on the deterministic cover time for common graphs

We now demonstrate how to apply above general results to obtain upper bounds for the edge cover time of the deterministic walk for many common graphs. As the general bounds Theorems 3.5, 3.10 and 3.13 all have a linear dependency on $\tilde{\kappa}$, the following upper bounds can be also stated depending on $\tilde{\kappa}$. However, for clarity we assume $\tilde{\kappa} = \mathcal{O}(1)$ here.

Theorem 3.14. *For complete graphs, $\widetilde{\text{EC}}(G) = \mathcal{O}(n^2)$.*

Proof. To bound the local divergence $\Psi(\mathbf{P}')$, observe that for any $t \geq 1$, $\mathbf{P}'^t_{i,j} = 1/n$ for every pair i, j . Hence we obtain

$$\Psi(\mathbf{P}') = \max_{v \in V} \left\{ \sum_{t=0}^{\infty} \sum_{\{i,j\} \in E} |\mathbf{P}'^t_{v,i} - \mathbf{P}'^t_{v,j}| \right\} = \max_{v \in V} \left\{ \sum_{\{i,j\} \in E} |\mathbf{P}'^0_{v,i} - \mathbf{P}'^0_{v,j}| \right\} = n - 1.$$

Plugging this into Theorem 3.8 yields the claim. \square

Theorem 3.15. *For d -dimensional torus graphs ($d \geq 1$ constant), $\widetilde{\text{EC}}(G) = \mathcal{O}(n^{1+1/d})$.*

Proof. Also here, we apply Theorem 3.5 and use the bound from [47, Thm. 8] that $\Psi(\mathbf{P}) = \mathcal{O}(n^{1/d})$. It is known that for $d = 1$, $\max_{u,v \in V} H(u, v) = \Theta(n^2)$, $d = 2$, $\max_{u,v \in V} H(u, v) = \Theta(n \log n)$ and for $d \geq 3$, $\max_{u,v \in V} H(u, v) = \Theta(n)$ (e.g., [13]). Hence the claim follows by Theorem 3.5. \square

Theorem 3.16. *For hypercubes, $\widetilde{\text{EC}}(G) = \mathcal{O}(n \log^2 n)$.*

Proof. To apply Theorem 3.13, we use the strong symmetry of the hypercube $H = H_{\log n}$. More precisely, we use the distance transitivity of the hypercube (cf. [10]), that is, for all vertices $w, x, y, z \in V$ with $\text{dist}(w, x) = \text{dist}(y, z)$ there is a permutation $\sigma: V \rightarrow V$ with $\sigma(w) = y$, $\sigma(x) = z$ and for all $u, v \in V$, $\{u, v\} \in E \Leftrightarrow \{\sigma(u), \sigma(v)\} \in E$.

We proceed to upper bound $\sum_{\{i,j\} \in E} |f_s(i, j)|$, where f_s is defined as in Definition 3.12. As one might expect, for distance-transitive graphs the ℓ_2 -minimal flow is distributing the flow uniformly among all edges connecting pairs of vertices with a different distance to s . More formally, [44, Thm. 5] showed that for any two vertices $i, j \in V$ with $i \in \Gamma^d(s)$ and $j \in \Gamma^{d+1}(s)$,

$$|f_s(i, j)| = \frac{1}{|E(\Gamma^d(s), \Gamma^{d+1}(s))|} \sum_{\ell=d+1}^{\log n} |\Gamma^\ell(s)|.$$

With $|\Gamma^\ell(s)| = \binom{\log n}{\ell}$ for all $0 \leq \ell \leq \log n$,

$$\sum_{\{i,j\} \in E} |f_s(i, j)| = \sum_{d=0}^{\log n-1} \sum_{\ell=d+1}^{\log n} \binom{\log n}{\ell} = \sum_{\ell=1}^{\log n} \ell \binom{\log n}{\ell} = \log n \cdot 2^{\log n-1} = \log n (n/2).$$

Moreover, it is a well-known result that on hypercubes, $\max_{u,v \in V} H(u, v) = \mathcal{O}(n)$ [1, p. 372]. Plugging this into Theorem 3.13 yields the claim. \square

Theorem 3.17. *For k -ary trees ($k \geq 2$ constant), $\widetilde{\text{EC}}(G) = \mathcal{O}(n \log n)$.*

Proof. We examine a complete k -ary tree ($k \geq 2$) of depth $\log_k n - 1 \in \mathbb{N}$ (the root has depth 0) where the number of nodes is $\sum_{i=0}^{\log_k n-1} k^i = n - 1$. To apply Theorem 3.13, we observe that on a cycle-free graph an ℓ_2 -minimal flow f is routed via shortest paths. Let us first assume that the distinguished node s of Definition 3.12 is the root and bound the corresponding optimal flow f_1 . In this case, $f_1(x, y) = k^{\log_k n - i - 1} - 1$ for $x \in \Gamma^i(s)$ and $y \in \Gamma^{i+1}(s)$. Hence,

$$\sum_{\{i,j\} \in E} |f_1(i, j)| = \sum_{d=0}^{\log_k n-1} k^d (k^{\log_k n - d - 1} - 1) = \sum_{d=0}^{\log_k n-1} \left(\frac{n}{k} - k^d \right) \leq \frac{n}{k} \log_k n.$$

Consider now the more general case, where the distinguished vertex s is an arbitrary vertex. Here the optimal flow f can be described as a superposition of a flow f_1 and f_2 , where f_1 sends a flow of n tokens from s to the root and f_2 sends $n - 1$ tokens from the root to all other vertices. Clearly, $\sum_{\{i,j\} \in E} |f_2(i, j)| \leq n \log_k n$ as a flow amount of n is routed over at most $\log_k n$ vertices. Therefore,

$$\sum_{\{i,j\} \in E} |f(i, j)| = \sum_{\{i,j\} \in E} |f_1(i, j) + f_2(i, j)| \leq \sum_{\{i,j\} \in E} |f_1(i, j)| + \sum_{\{i,j\} \in E} |f_2(i, j)| \leq \frac{k+1}{k} n \log_k n$$

Moreover, we know from [55, Proof of Corollary 9] that $\max_{u,v} H(u, v) = \mathcal{O}(n \log_k n)$. Hence applying Theorem 3.13 yields the claim. \square

Theorem 3.18. *For lollipop graphs, $\widetilde{\text{EC}}(G) = \mathcal{O}(n^3)$.*

Proof. We use the following strengthened version of Theorem 3.13 (see last line of the proof of Theorem 3.13),

$$\widetilde{\mathbf{EC}}_v(G) \leq \max_{u \in V} \mathbf{H}'(u, v) + \Delta n + \Delta \sum_{\{i, j\} \in E} |f_v(i, j)|, \quad (9)$$

where $\widetilde{\mathbf{EC}}_v(G)$ refers to a random walk that starts at the vertex v . Note that to apply equation (9), we have to consider a random walk with transition matrix $\mathbf{P}' = \mathbf{I} - \frac{1}{n+1} \mathbf{L}$ and hitting times $\mathbf{H}'(\cdot, \cdot)$.

We first argue why it is sufficient to consider the case where the deterministic walk starts at vertex $v = n/2$. First, if the deterministic walk starts at any other vertex in the complete graph, we know from our upper bound on the deterministic cover time on complete graphs (Theorem 3.14) that after $\mathcal{O}(n^2)$ steps, the vertex $n/2$ is reached. Similarly, we know from Theorem 3.15 that if the random walk starts at any point of the path, it reaches the vertex $n/2$ within $\mathcal{O}(n^3)$ steps (note the extra factor of $\mathcal{O}(n)$, as in the corresponding deterministic walk model to \mathbf{P}' , each node on the path has $n/2 + 1$ loops).

So let us consider a random walk that starts at vertex $v = n/2$. To apply equation (9), we have to bound $\sum_{\{i, j\} \in E} |f_v(i, j)|$ for a ℓ_2 -optimal flow that sends a flow amount of one from vertex $n/2$ to all other vertices (cf. Definition 3.12).

Clearly, the ℓ_2 -optimal flow sends at each edge $\{i-1, i\} \in E$, $n/2 < i < n$ in the path a flow of $n-i$. Moreover, it assigns to each edge $(i, n/2)$ with $1 \leq i \leq n/2-1$ a flow of 1. Hence,

$$\sum_{\{i, j\} \in E} |f_v(i, j)| = (n/2 - 1) \cdot 1 + \sum_{i=n/2+1}^n i = \mathcal{O}(n^2).$$

Our final step is to prove $\max_{u, v \in V} \mathbf{H}'(u, v) = \mathcal{O}(n^3)$ for the symmetric random walk. In fact, we shall prove that this holds for arbitrary graphs. Note that by the symmetry of the transition matrix, $\mathbf{H}'(u, u) = 1/(\pi_u) = n$. So take a shortest path $\mathcal{P} = (u_1 = u, u_2, \dots, u_\ell = v)$ of length ℓ between u and v in G . Note that each time the walk is at any vertex u_i it moves to the vertex u_{i+1} with probability $1/(\Delta+1)$. Hence if $\tau'(u, v)$ describes the random variable for the first hit of v when starting from u , we have for any $1 \leq j \leq \ell-1$,

$$\tau'(u_j, u_{j+1}) = 1 + \sum_{i=1}^{\text{Geo}(1/(\Delta+1))-1} X_i,$$

where X_i is the intermediate time between the i -th and $(i+1)$ -th visit of u_j . Since all X_i are independent and identically distributed random variables with expectation n , we can apply Walds equation [52] to get

$$\mathbf{H}'(u_j, u_{j+1}) = \mathbf{E}[\tau'(u_j, u_{j+1})] = 1 + \left(\mathbf{E} \left[\text{Geo} \left(\frac{1}{\Delta+1} \right) \right] - 1 \right) \cdot \mathbf{E}[X_i] = 1 + \Delta \cdot n.$$

Now using the triangle inequality, we finally get

$$\mathbf{H}'(u_1, u_\ell) \leq \sum_{j=1}^{\ell-1} \mathbf{H}'(u_j, u_{j+1}) \leq (\ell-1) \cdot (1 + \Delta \cdot n) = \mathcal{O}(n^3).$$

Plugging in our findings in equation (9), the claim follows. \square

The last theorem about the lollipop graph (a graph that consists of a clique with $n/2$ vertices connected to a path of length $n/2$) might look weak, but turns out to be tight as we will show in Theorem 4.4.

4 Lower Bounds on the Deterministic Cover Time

We first prove a general lower bound of $\Omega(m)$ on the deterministic cover time for all graphs. Afterwards, for all graphs examined in Section 3.5 for which this general bound is not tight (cycle, path, tree, torus, hypercube, expander) we present stronger lower bounds which match their respective upper bounds.

Theorem 4.1. *For all graphs, $\widetilde{VC}(G) \geq m - \delta$.*

Proof. Let w be a vertex in G with minimum degree δ . Consider the graph $G \setminus \{w\}$ with each undirected edge $\{u, v\}$ replaced by a two directed edges (u, v) and (v, u) . Then there is an Euler tour through $G \setminus \{w\}$. We now choose the rotor sequence $(\tilde{s}(u, 1), \tilde{s}(u, 2), \dots, \tilde{s}(u, \deg(u)))$ of a vertex $u \in V \setminus \{w\}$ according to the order the neighbors of u are visited by the Euler tour. Then the deterministic walk takes the whole Euler tour through $G \setminus \{w\}$ of length $m - \deg(w) = m - \delta$ before visiting w . \square

As a telling example for a lower bound of the deterministic cover time of a simple graph, let us examine a rooted complete k -ary tree (k constant). We choose the rotors to move clockwise and let the walk start at the root. It is then easy to observe that a configuration where all rotors initially point downwards towards their respective rightmost successor leads to a order of explored vertices corresponding to a depth-first-search. By definition of $\widetilde{VC}(G)$ this only implies a trivial lower bound for the deterministic cover time of $\Omega(n)$. Analogously, a configuration where all rotors initially point towards the root leads to a order of the explored vertices corresponding to a breath-first-search. However, an easy calculation also just gives a linear bound for this walk.

Now consider the following initial configuration: each vertex in the leftmost subtree of the root is pointing upwards, each vertex in the other subtrees downwards and the root vertex is pointing to the rightmost subtree. Then every time the deterministic walk enters one of the subtrees where the rotors are pointing downwards, it does a depth-first-search walk of length $\Theta(n)$. When it reaches the root again, all inner vertices of the subtree are visited k times and all rotors are pointing downwards again. On the other hand, when the deterministic walk enters the leftmost subtree where all rotors are pointing upwards, it only visits one more level than it did in the previous visit corresponding to a breath-first-search. Overall, the leftmost tree is visited $\log_k n$ times and all other vertices are visited between two visits of the leftmost tree. This gives a tight lower bound of $\Omega(n \log n)$ and the following theorem.

Theorem 4.2. *For k -ary trees ($k \geq 2$ constant), $\widetilde{VC}(G) = \Omega(n \log n)$.*

A similar analysis gives the following asymptotically tight lower bounds.

Theorem 4.3. *For cycles, $\widetilde{VC}(G) = \Omega(n^2)$.*

Proof. Let the $n + 1$ vertices of an odd cycle be numbered consecutively from $-n/2$ to $n/2$. Consider the initial configuration where every rotor is pointing towards the vertex's neighbor with a smaller number in absolute value and the rotor of 0 point towards 1. Assume that

the walk starts from vertex 0. It is easy to see that the sequence of visited vertices by the walk consists of $n/2$ phases where phase i with $1 \leq i < n/2$ is of length $4i$ and visits $0, 1, 2, \dots, i-1, i, i-1, \dots, 2, 1, 0, -1, -2, \dots, -(i-1), -i, -(i-1), \dots, -3, -2, -1$ while the last phase visits $0, 1, 2, \dots, n/2-1, n/2, n/2-1, \dots, 2, 1, 0, -1, -2, \dots, -(n/2-1), -n/2$. Thus $(n^2 + n)/2$ steps are required to cover all vertices. Note that the same argument gives a lower bound of $(n-1)^2 + 1$ steps for the path. \square

Theorem 4.4. *For lollipop graphs, $\widetilde{VC}(G) = \Omega(n^3)$.*

Proof. We number the vertices in the clique consecutively from 1 to $n/2$, and the vertices on the path consecutively from $n/2 + 1$ to n . Further assume that the vertices $n/2$ and $n/2 + 1$ are connected. Consider the following configuration: each vertex on the path is pointing towards the vertex with a smaller number, and the rotor's permutation of the vertices in the complete graph are chosen such that a walk starting from the complete graph takes an Eulerian tour therein before escaping to the path. We know from the proof of Theorem 4.3 that the root vertex is visited n times, before the walk reaches the endpoint n . Since everytime the walk returns to the complete graph, it takes a complete Euler tour of length $\Theta(n^2)$ there, the theorem follows. \square

More involved techniques are necessary for expanders, tori and hypercubes.

Theorem 4.5. *There are expander graphs with $\widetilde{VC}(G) = \Omega(n \log n)$.*

To prove Theorem 4.5 we first state the following property of deterministic walks of Priezzhev et al. [46].

Lemma 4.6 (Priezzhev et al. [46, p. 5080]). *Between two successive visits of the same directed edge the unweighted deterministic walk visits no other directed edge twice.*

Proof. Let the deterministic walk visit $\tilde{x}_0, \tilde{x}_1, \dots, \tilde{x}_t, \tilde{x}_{t+1}$. We assume that the last edge $(\tilde{x}_t, \tilde{x}_{t+1})$ is equal to the first edge $(\tilde{x}_0, \tilde{x}_1)$ and this edge is not visited in between. Seeking a contradiction, we assume that there is an edge (u, v) which is the first edge that is visited twice in between, that is, there are (minimal) times i, j with $0 < i < j < t$ such that $(u, v) = (\tilde{x}_i, \tilde{x}_{i+1}) = (\tilde{x}_j, \tilde{x}_{j+1})$. If the rotor of u pointed twice towards v , the walk must have left u $\deg(u) + 1$ often. Hence the walk must also have entered u that often. As there are only $\deg(u)$ edges going from any vertex to u , one of these edges must have visited twice, too. This contradicts our assumption that (u, v) was the first edge visited twice. \square

Proof of Theorem 4.5. We construct an expander $G = (V, E)$ with expansion constant $1/20$ and prove $\widetilde{VC}(G) = \Omega(n \log n)$. G consists of two subgraphs G_{ex} and G_{tr} . $G_{\text{ex}} = (V_{\text{ex}}, E_{\text{ex}})$ is a d -regular ($d \geq 10$ is a sufficiently large constant) expander graph with expansion constant $7/8$ and $n/2$ vertices. $G_{\text{tr}} = (V_{\text{tr}}, E_{\text{tr}})$ is a tree with $n/2$ leaves, where the root has d successors and all other nodes besides leaves have $d - 1$ successors. Let E_{ma} be the union of d perfect matchings between the leaves of V_{tr} and V_{ex} .

Then $V = V_{\text{ex}} \cup V_{\text{tr}}$ and $E = E_{\text{ex}} \cup E_{\text{tr}} \cup E_{\text{ma}}$.

We first prove that such a graph exists, prove some properties and that G is an expander itself. At the end we prove the bound for $\widetilde{VC}(G)$.

We choose G_{ex} as a d -regular Ramanujan graph with expansion constant at least $7/8$ and $n/2$ vertices i.e., $|E(X, X^c)| \geq \frac{7}{8}d|X|$, for all $X \subseteq V_{\text{ex}}$ with $1 \leq |X| \leq n/4$. Such a graph exists since for random d -regular graphs, $\lambda_2 = \mathcal{O}(d^{-1/2})$ [24] and moreover $|E(X, X^c)|/(d|X|) \geq \sqrt{1 - \lambda_2} \geq \sqrt{1 - \mathcal{O}(d^{-1/2})}$.

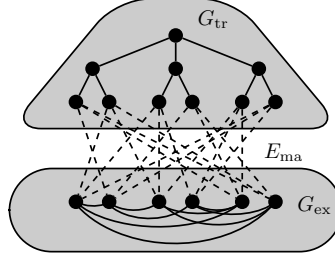


Figure 1: An illustration of the expander graph G used in Theorem 4.5 for $d = 3$ and $n = 12$.

Let us now consider a set $X \subseteq V_{\text{ex}}$ with $(1/4)n \leq |X| \leq (3/8)n$. Then,

$$|E(X, X^c)| \geq \frac{7}{8}d\frac{1}{4}n - d\left(|X| - \frac{1}{4}n\right) \geq \frac{7}{32}dn - \frac{1}{8}dn = \frac{3}{32}dn \geq \frac{3}{32}d\frac{8}{3}|X| = \frac{1}{4}d|X|. \quad (10)$$

To calculate $|V|$, observe that the total number of vertices in G_{tr} is

$$|V_{\text{tr}}| \leq \sum_{i=0}^{\log_d(n/2)} d^i = \frac{d(n/2) - 1}{d - 1} \leq \frac{d}{d - 1} \frac{n}{2} \leq \frac{9}{8} \frac{n}{2} = \frac{9}{16}n$$

and therefore G has $|V| \leq (17/16)n \leq (9/8)n$ vertices with $\delta(G) = d$ and $\Delta(G) = 2d$. To see that G is also an expander graph, take any subset $X \subseteq V$ with $1 \leq |X| \leq (9/16)n$. Let $X_{\text{tr}} := X \cap V_{\text{tr}}$ and $X_{\text{ex}} := X \cap V_{\text{ex}}$.

- (i) Consider first the case where $|X_{\text{tr}}| \geq 4|X_{\text{ex}}|$. Observe that for any X_{tr} , $|E(X_{\text{tr}}, X_{\text{tr}})| \leq |X_{\text{tr}}| - 1$ as G_{tr} is a tree. Therefore,

$$\begin{aligned} |E(X, X^c)| &\geq |E(X_{\text{tr}}, X^c)| = |E(X_{\text{tr}}, V)| - |E(X_{\text{tr}}, X_{\text{tr}})| - |E(X_{\text{tr}}, X_{\text{ex}})| \\ &\geq d|X_{\text{tr}}| - |X_{\text{tr}}| - d|X_{\text{ex}}| \geq d|X_{\text{tr}}| - |X_{\text{tr}}| - \frac{d}{4}|X_{\text{tr}}| \\ &\geq \left(\frac{3}{4}d - 1\right)|X_{\text{tr}}| \geq \left(\frac{3}{4}d - 1\right)\frac{4}{5}|X| \geq \frac{13}{25}d|X|. \end{aligned}$$

- (ii) Assume now that $|X_{\text{tr}}| \leq 4|X_{\text{ex}}|$. If $|X_{\text{ex}}| \leq (3/8)n$, then equation (10) implies that

$$|E(X, X^c)| \geq |E(X_{\text{ex}}, V_{\text{ex}} \setminus X)| = |E(X_{\text{ex}}, V_{\text{ex}} \setminus X_{\text{ex}})| \geq \frac{1}{4}d|X_{\text{ex}}| \geq \frac{1}{20}d|X|.$$

On the other hand, if $|X_{\text{ex}}| \geq (3/8)n$, it follows that $|X_{\text{tr}}| \leq (3/16)n$. Since each vertex in X_{ex} has d edges to V_{tr} , we have

$$\begin{aligned} |E(X, X^c)| &\geq |E(X_{\text{ex}}, X^c)| \geq |E(X_{\text{ex}}, X_{\text{ex}}^c)| - |E(X_{\text{ex}}, X_{\text{tr}})| \geq \frac{3}{8}dn - \frac{3}{16}(d+1)n \\ &= \frac{3}{16}(d-1)n \geq \frac{3}{16}(d-1)\frac{16}{9}|X| = \frac{1}{3}(d-1)|X| \geq \frac{3}{10}d|X|. \end{aligned}$$

Hence we conclude that the graph G is an expander graph with expansion constant $1/20$.

Phase	Steps	Situation at the end of the respective phase						Corresponding Figure
		C_1	C_2	C_3	C_4	C_5	C_6	
1	1	<u>in</u>	in	in	in	in	in	Figure 2(a)
2	2–9	<u>cycle</u>	in	in	in	in	in	Figure 2(b)
3	10–18	out	<u>in</u>	in	in	in	in	Figure 2(c)
4	19–49	<u>in</u>	cycle	in	in	in	in	Figure 2(d)
5	50–57	<u>cycle</u>	cycle	in	in	in	in	Figure 2(e)
6	58–66	out	<u>cycle</u>	in	in	in	in	Figure 2(f)
7	67–83	out	out	<u>in</u>	in	in	in	Figure 2(g)
8	84–138	out	<u>in</u>	cycle	in	in	in	Figure 2(h)
9	139–169	<u>in</u>	cycle	cycle	in	in	in	Figure 2(i)
10	170–177	<u>cycle</u>	cycle	cycle	in	in	in	Figure 2(j)
11	178–186	out	<u>cycle</u>	cycle	in	in	in	Figure 2(k)
12	187–203	out	out	cycle	in	in	in	Figure 2(l)

Table 3: First twelve phases of a deterministic walk on the two-dimensional torus. A visual description of the phases on the 7×7 torus is given in Figure 2. The different states are defined in the proof of Theorem 4.7. Underlined states indicate the last position of the deterministic walk.

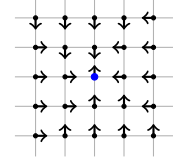
We are now ready to define the rotors. As in the proof of Theorem 4.1, choose an Euler tour of the directed graph G_{ex} and set the rotors of V_{ex} and the initial position such that the deterministic walk on G first performs an Euler tour on G_{ex} before visiting any node from V_{tr} . For vertices from V_{tr} we choose the rotor sequence similar to the proof of Theorem 4.2 such that the direction of the root is always the last one in the sequence. Let $u_i \in V_{\text{tr}}$ be the first node in level i with $0 < i < \log_d(n/2) - 2$ of the tree G_{tr} which is reached. Let this happen at time t_i from a node u_{i+1} in level $i + 1$. Let t_{i+1} be the first time u_{i+1} is visited. By choice of the rotor sequence, only at the $(d + 1)$ -th visit to u_{i+1} its rotor can point upwards to u_i . As u_{i+1} has only d children, one child u_{i+2} must be visited twice between times t_{i+1} and t_i . Hence also the directed edge (u_{i+2}, u_{i+1}) is visited twice in this time interval. Assume there was an edge $e \in E_{\text{ex}}$ which was not visited in this time interval. We know that this edge e is visited before time t_{i+1} by the Euler tour and that it is visited after time t_i as the graph is strongly connected and the deterministic walk eventually visits all edges arbitrarily often. Lemma 4.6 implies that then e must also be visited between times t_{i+1} and t_i . Overall, between every new level of V_{tr} which is explored, the deterministic walk has to visit all edges E_{ex} . Hence it takes $\Omega(n \log n)$ steps to visits all vertices of G . \square

Theorem 4.7. *For two-dimensional torus graphs, $\widetilde{VC}(G) = \Omega(n^{3/2})$.*

Proof. Consider a two-dimensional $\sqrt{n} \times \sqrt{n}$ torus. For simplicity we assume that \sqrt{n} is an odd integer and represent the vertices by two coordinates (x, y) with $-L \leq x, y \leq L$ with $L := (\sqrt{n} - 1)/2$.

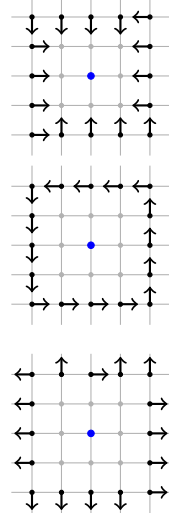
Let all rotor sequences be ordered clockwise (that is, $\uparrow, \rightarrow, \downarrow, \leftarrow, \dots$) and start with a rotor in the direction of the origin $(0, 0)$. More precisely, let the initial rotor direction at vertex (x, y) be

- (i) \uparrow if $y \leq -1$ and $y \leq -x$ and $y < x$ or $(x, y) = (0, 0)$,
- (ii) \rightarrow if $x \leq -1$ and $-x > y$ and $x \leq y$,
- (iii) \downarrow if $y \geq 1$ and $y \geq -x$ and $y > x$,
- (iv) \leftarrow if $x \geq 1$ and $-x < y$ and $x \geq y$.



We will start the random walk from the origin $(0, 0)$. Denote by $C_i := \{(x, y) \in V : \max\{x, y\} = i\}$ the boundary of the square defined by the corners (i, i) , $(i, -i)$, $(-i, -i)$, $(-i, i)$. For each square C_i , $1 \leq i \leq L$, we define three different states called in, cycle and out:

$$\begin{aligned}
 \text{(i) } C_i \text{ is in, iff } \rho(x, y) &= \begin{cases} \downarrow & \text{for } y = i \text{ and } -i \leq x < i, \\ \leftarrow & \text{for } x = i \text{ and } -i < y \leq i, \\ \uparrow & \text{for } y = -i \text{ and } -i < x \leq i, \\ \rightarrow & \text{for } x = -i \text{ and } -i \leq y < i. \end{cases} \\
 \text{(ii) } C_i \text{ is cycle, iff } \rho(x, y) &= \begin{cases} \downarrow & \text{for } x = -i \text{ and } -i < y \leq i, \\ \leftarrow & \text{for } y = i \text{ and } -i < x \leq i, \\ \uparrow & \text{for } x = i \text{ and } -i \leq y < i, \\ \rightarrow & \text{for } y = -i \text{ and } -i \leq x < i. \end{cases} \\
 \text{(iii) } C_i \text{ is out, iff } \rho(x, y) &= \begin{cases} \leftarrow & \text{for } x = -i \text{ and } -i < y \leq i, \\ \uparrow & \text{for } y = i \text{ and } -i < x \leq i \text{ and } x \neq 0, \\ \rightarrow & \text{for } x = i \text{ and } -i \leq y < i \\ & \text{or } y = i \text{ and } x = 0, \\ \downarrow & \text{for } y = -i \text{ and } -i \leq x < i. \end{cases}
 \end{aligned}$$



By definition, the initial rotor directions of all squares C_i , $1 \leq i \leq L$, is in. We decompose the deterministic walk in phases such that after each phase the walk is at a vertex $(0, y)$ for some y with $1 \leq y \leq L$, every square C_i , $1 \leq i \leq L$, has a well-defined state (in, cycle or out), and the rotor at $(0, 0)$ is \rightarrow . The first phase has length one. Hence after the first phase the rotors are $(C_1, C_2, C_3, \dots) = (\underline{\text{in}}, \text{in}, \text{in}, \dots)$ where the underline in $\underline{\text{in}}$ marks the current position of the walk. We now observe the following three simple rules which can be easily proven by induction.

- (i) If the walk is at $(0, 1)$ and $C_1 = \text{in}$, then after eight steps the walk is at $(0, 2)$ and $C_1 = \text{cycle}$. Or in short: $(\underline{\text{in}}, \dots) \xrightarrow{8} (\underline{\text{cycle}}, \dots)$.
- (ii) If the walk is at $(0, y)$, $y \geq 1$, and $C_y = \text{cycle}$, then after $8y + 1$ steps the walk is at $(0, y + 1)$ and $C_y = \text{out}$. Or in short: $(\dots, \underline{\text{cycle}}, C_{y+1}, \dots) \xrightarrow{8y+1} (\dots, \underline{\text{out}}, C_{y+1}, \dots)$.
- (iii) If the walk is at $(0, y)$, $y \geq 2$, and $C_y = \text{in}$ as well as $C_{y-1} = \text{out}$, then after $24y - 17$ steps the walk is at $(0, y - 1)$ and $C_y = \text{cycle}$ as well as $C_{y-1} = \text{in}$. Or in short: $(\dots, C_{y-1}, \underline{\text{in}}, \dots) \xrightarrow{24y-17} (\dots, \underline{\text{in}}, \text{cycle}, \dots)$.

Table 3 shows the first twelve phases applying above three rules. In the introduced short notation, after the first phase the rotors are $(\underline{\text{in}}, \text{in}^{L-1})$. The second phase applies rule (i) and reaches $(\underline{\text{in}}, \text{in}^{L-1}) \xrightarrow{8} (\underline{\text{cycle}}, \text{in}^{L-1})$. The three subsequent phases can be described as

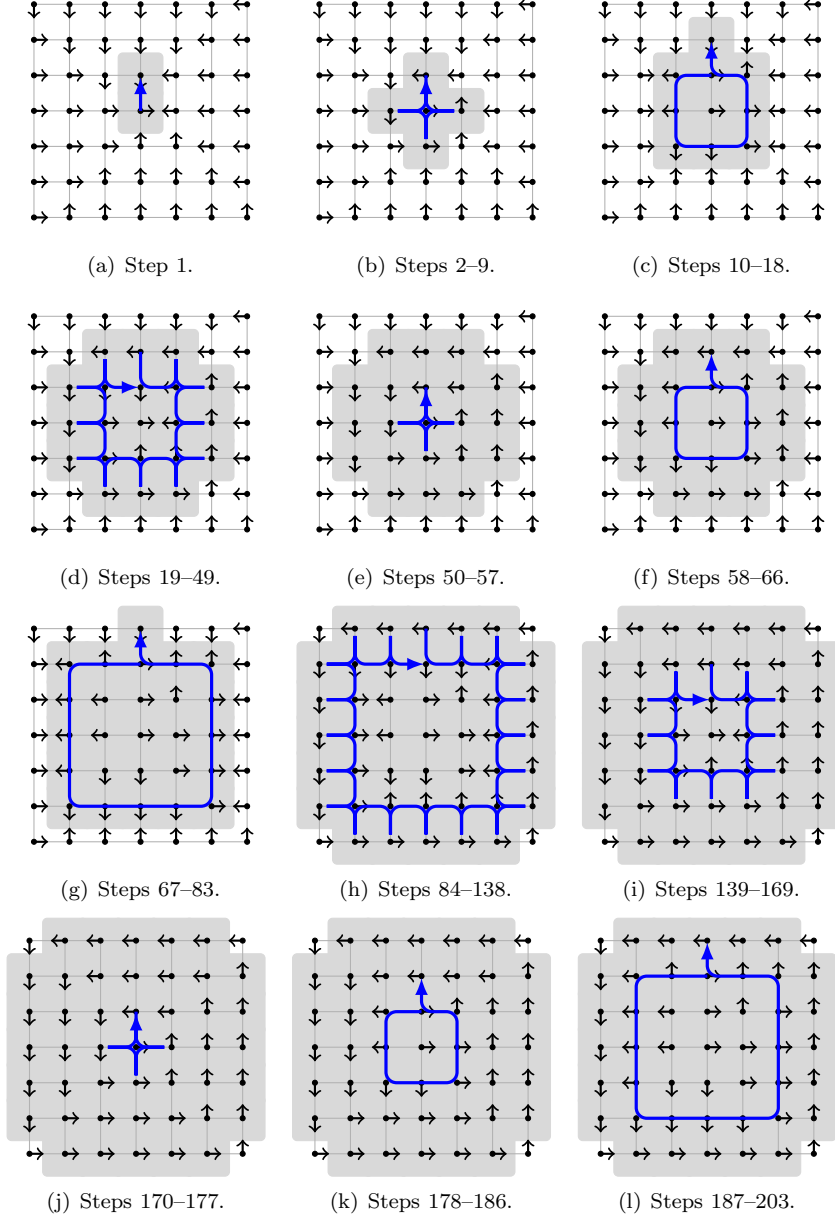


Figure 2: The first eleven phases of the deterministic walk on the two-dimensional 7×7 torus. All rotors are initially pointing towards the origin. In each phase the deterministic walk is shown as a blue arrow. The depicted rotors correspond to the rotor directions at the end of the respective phase. The gray shaded area marks all covered vertices at this time.

follows (corresponding to Figure 2 (b)–(e)):

$$(\underline{\text{cycle}}, \text{in}^{L-1}) \xrightarrow{9} (\text{out}, \underline{\text{in}}, \text{in}^{L-2}) \xrightarrow{31} (\underline{\text{in}}, \text{cycle}, \text{in}^{L-2}) \xrightarrow{8} (\underline{\text{cycle}}, \text{cycle}, \text{in}^{L-2}).$$

These three phases (or 48 steps) already reveal the general pattern. By induction one can prove for all k with $1 \leq k < L$:

$$\begin{aligned} (\underline{\text{cycle}}, \text{cycle}^{k-1}, \text{in}^{L-k}) &\xrightarrow{\sum_{y=1}^k (8y+1)} (\text{out}^k, \underline{\text{in}}, \text{in}^{L-k-1}) \xrightarrow{\sum_{y=2}^{k+1} (24y-17)} (\underline{\text{in}}, \text{cycle}^k, \text{in}^{L-k-1}) \\ &\xrightarrow{8} (\underline{\text{cycle}}, \text{cycle}^k, \text{in}^{L-k-1}). \end{aligned}$$

This shows that the deterministic walk needs $\sum_{y=1}^k (8y+1) + \sum_{y=2}^{k+1} (24y-17) + 8 = 8(2k+1)(k+1)$ steps to go from $(\underline{\text{cycle}}, \text{cycle}^{k-1}, \text{in}^{L-k})$ to $(\underline{\text{cycle}}, \text{cycle}^k, \text{in}^{L-k-1})$. To get a lower bound on the deterministic cover time, we bound the time to reach $(0, L)$ with $C_L = \text{cycle}$:

$$\begin{aligned} (\underline{\text{in}}, \text{in}^{L-1}) &\xrightarrow{8} (\underline{\text{cycle}}, \text{in}^{L-1}) \xrightarrow{\sum_{k=1}^{L-1} 8(2k+1)(k+1)} (\underline{\text{cycle}}, \text{cycle}^{L-1}) \\ &\xrightarrow{\sum_{y=1}^{L-1} 8y+1} (\text{cycle}^{L-1}, \underline{\text{cycle}}). \end{aligned}$$

After $(\text{cycle}^{L-1}, \underline{\text{cycle}})$ is reached, the deterministic walk only needs $7L$ further steps to go from $(0, L)$ along $C_L = \text{cycle}$ to the last uncovered vertex (L, L) . This gives an overall lower bound for the deterministic cover time of

$$1 + 8 + \sum_{k=1}^{L-1} 8(2k+1)(k+1) + \sum_{y=1}^{L-1} 8y+1 + 7L = \frac{16}{3}L^3 + 8L^2 + \frac{8}{3}L = \frac{2}{3}(n^{3/2} - \sqrt{n}). \quad \square$$

Theorem 4.8. *For hypercubes, $\widetilde{\text{VC}}(G) = \Omega(n \log^2 n)$.*

Proof. We consider the d -dimensional hypercube with $n = 2^d$ vertices corresponding to bitstrings $\{0, 1\}^d$. A pair of vertices is connected if their bitstrings differ in exactly one bit.

We first note that if all rotors are set up the same way (w.r.t. the dimension), all vertices of the hypercube are covered within $\mathcal{O}(n)$ steps. Hence we choose the rotor sequences differently, that is, let all rotor sequences be sorted lexicographically and let the walk start at 0^d . We bound the time to reach 1^d . The chosen rotor sequence implies that every vertex x first visits the neighboring vertices y with $|y|_1 < |x|_1$ and then the neighboring vertices y with $|y|_1 > |x|_1$. The resulting walk can be nicely described as a sequence of depth first searches (DFS) on a tree T_d which is defined as follows. The vertices are bitstrings $\{0, 1\}^d$ corresponding to the vertices of H_d . If there is an edge $\{u, v\}$ in T_d , then $\{u, v\}$ is an edge in H_d , too (but not the other way around). The root of T_d is 0^d . In level i there are only vertices x with $|x|_1 = i$. Every vertex x has either 0 or $|x|_0$ children corresponding to neighbors y of x with $|y|_1 = |x|_1 + 1$. The root has d children. All other vertices x have $|x|_0$ children iff the single bit in which it differs from its parent is left of the leftmost 1-bit. That is, the tree is truncated if the bit which is flipped from the parent to the child is not a leading zero. Note that this implies that on level i there are i copies of each vertex with $|x|_1 = i$ in T_d and only the leftmost recurses to the next level. Figure 3 on page 21 shows the deterministic walk on H_5 and the corresponding tree T_5 as defined above.

We decompose the deterministic walk on H_d in d phases such that phase i has length $2 \sum_{j=1}^i j \binom{d}{j}$ and show the the following:

- (i) Initially, the rotors corresponding to the vertices in T_d point towards their respective parent.

- (ii) After the i -th phase, all rotors of vertices x with $|x|_1 > i$ point towards their respective parent in T_d while all rotors of vertices x with $|x|_1 \leq i$ point to their leftmost child in T_d , i.e., to their lexicographic smallest neighbor with one more bit set to one.
- (iii) The i -th phase of the deterministic walk on H_d visits the same vertices in the same order as a DFS on T_d with limited depth i .
- (iv) T_d has $i \binom{d}{i}$ vertices on level i with $i > 0$.

(i) holds by definition of T_d for the chosen rotor sequence. We now prove simultaneously (ii) and (iii) by induction. For the first phase of $2d$ steps it is easy to see as it alternates between 0^d and all nodes x with $|x|_1 = 1$ (in increasing order). This phase ends at the root 0^d whose rotor then points at $0^{d-1}1$ (as initially). The rotors of nodes x with $|x|_1 = 1$ now point to the lexicographically smallest neighbor y with $|y|_1 = 2$. Note that for every vertex y with $|y|_1 = 2$ there are two vertices x with $|x|_1 = 1$ whose rotor points to y . Let us now assume (ii) and (iii) holds after the i -th phase. Then all rotors of vertices x with $|x|_1 \leq i$ point downwards in T_d to their leftmost child which is also the lexicographically smallest. It is obvious that then the deterministic walk on H_d exactly performs a DFS of limited depth $i + 1$ on H_d up to the point when a vertex is visited the second time within this phase. If this vertex x is in the last visited layer, i.e., $|x| = i + 1$, then its rotor points upwards and the DFS is not disturbed. If a vertex x with $|x| \leq i$ is visited the second time, then by definition of T_d , this vertex has no children. Hence the DFS in T_d goes back to its parent which is the same vertex to which the deterministic walk in H_d moves as the rotor is already pointing at its second neighbor in its rotor sequence. The same holds for the third, fourth, and so on visit of a vertex. Overall, this $(i + 1)$ -th phase visits all vertices in T_d up to depth $(i + 1)$ and changes the rotors of vertices x with $|x|_1 = (i + 1)$ downwards. This proves (ii) and (iii). (iv) immediately follows from the fact that there are i copies of each vertex x with $|x|_1 = i$ on level i in T_d .

The number of vertices visited in phase i with $i < d$ is twice the number of edges up to depth i . Therefore the length of phase $i < d$ is $2 \sum_{j=1}^i j \binom{d}{j}$ and the total length of the deterministic walk on H_d until 1^d is discovered is

$$\begin{aligned}
d + 1 + 2 \sum_{i=0}^{d-1} \sum_{j=1}^i j \binom{d}{j} &= d + 1 + 2 \sum_{j=0}^d j (d - j) \binom{d}{j} = d + 1 + 2d \sum_{j=0}^{d-1} j \binom{d-1}{j} \\
&= d + 1 + d(d - 1) 2^{d-1} = (n \log^2 n)/2 + \mathcal{O}(n \log n). \quad \square
\end{aligned}$$

5 Short Term Behavior

For random walks, Barnes and Feige [7, 8] examined how fast a random walk covers a certain number of vertices and/or edges. Table 4 provides an overview of their bounds compared to ours. For the deterministic walk, we can show the following result about the rate at which the walk discovers new edges in the short term.

Theorem 5.1. *All deterministic walks with $\tilde{\kappa} = 1$ visit \mathcal{N} distinct vertices within $\min \{ \mathcal{O}(\mathcal{N}\Delta + (\mathcal{N}\Delta/\delta)^2), \mathcal{O}(m + (m/\delta)^2) \}$ steps and \mathcal{M} distinct edges within $\mathcal{O}(\mathcal{M} + (\mathcal{M}/\delta)^2)$ steps.*

The proof of Theorem 5.1 is based on some combinatorial property of the deterministic walk. Let us first observe a simple graph-theoretic lemma.

	Random Walk	Deterministic Walk
time to visit \mathcal{N} vertices on arbitrary graphs	$\begin{cases} \mathcal{O}(\mathcal{N}^3) & [8, \text{Thm. 1.1}] \\ \mathcal{O}(m\mathcal{N}) & [8, \text{Thm. 1.4}] \end{cases}$	$\begin{cases} \mathcal{O}(\mathcal{N}\Delta + (\mathcal{N}\Delta/\delta)^2) & (\text{Thm. 5.1}) \\ \mathcal{O}(m + (m/\delta)^2) & (\text{Thm. 5.1}) \end{cases}$
time to visit \mathcal{M} edges on arbitrary graphs	$\begin{cases} \mathcal{O}(\mathcal{M}^2) & [8, \text{Thm. 1.2}] \\ \mathcal{O}(n\mathcal{M}) & [8, \text{Thm. 1.4}] \\ \mathcal{O}(\mathcal{M} + (\mathcal{M}^2 \log \mathcal{M})/\delta) & [7, \text{Thm. 5}] \end{cases}$	$\mathcal{O}(\mathcal{M} + (\mathcal{M}/\delta)^2) \quad (\text{Thm. 5.1})$

Table 4: Short term behavior of random and deterministic walk. For the time to cover \mathcal{N} vertices, the bounds for the random walk are always as good as the corresponding bounds for the deterministic walk. The two respective upper bounds for the time to cover \mathcal{M} edges are incomparable in general.

Lemma 5.2. *For any graph $G = (V, E)$, vertex $v \in V$, and $i \geq 0$ with $\Gamma^{i+1}(v) \neq \emptyset$, we have*

$$|E(\Gamma^i(v)) \cup E(\Gamma^{i+1}(v)) \cup E(\Gamma^{i+2}(v))| \geq \delta^2/6.$$

Proof. Fix a vertex $u \in \Gamma^{i+1}(v)$. Clearly, there exists a $j \in \{i, i+1, i+2\}$ such that $|E(u, \Gamma^j(v))| \geq \delta/3$. Since this implies that $|\Gamma^j(v)| \geq \delta/3$, we have $|E(\Gamma^j(v))| \geq |\Gamma^j(v)|\delta/2 \geq \delta^2/6$ and the claim follows. \square

In the proof of Theorem 5.1 we will also need the following property borrowed from Yanovski et al. [53].

Lemma 5.3. *For any time t and edges $\{u, v\}, \{v, w\} \in E$ it holds that*

$$|\tilde{N}_t(u \rightarrow v) - \tilde{N}_t(v \rightarrow w)| \leq 2.$$

Proof of Theorem 5.1. We start with the second claim. Assume that there is an edge $e = (u, v) \in E$ with $\tilde{N}_t(e) \geq 13\sqrt{t}/\delta$. Then we know that for all adjacent edges $e' = (v, w)$ that $\tilde{N}_t(e') \geq 13\sqrt{t}/\delta - 2$ as $|\tilde{N}_t(e) - \tilde{N}_t(e')| \leq 2$ by [53, Cor. 4]. More generally, for an edge $e = (x, y)$ with $\text{dist}(u, x) = i$ we have

$$\tilde{N}_t(x \rightarrow y) \geq 13\sqrt{t}/\delta - 2i. \quad (11)$$

As in the proof of the first claim, we may assume that $\Gamma^i(u) \neq \emptyset$ for $1 \leq i \leq 2\sqrt{t}/\delta$, as otherwise all edges have been traversed already by equation (11). With Lemma 5.2,

$$\begin{aligned}
t &= \sum_{\{u,v\} \in E} \tilde{N}_t(u \rightarrow v) \\
&\geq \sum_{k=0}^{2\sqrt{t}/(3\delta)} |E(\Gamma^{3k}(v)) \cup E(\Gamma^{3k+1}(v)) \cup E(\Gamma^{3k+2}(v))| (13\sqrt{t}/\delta - 6(k+1)) \\
&> \sum_{k=0}^{2\sqrt{t}/(3\delta)-1} \frac{\delta^2}{6} (13\sqrt{t}/\delta - 6(k+1)) = \frac{1}{6}\delta^2 \left(\frac{2\sqrt{t}}{3\delta} \right) 9 \frac{\sqrt{t}}{\delta} = t,
\end{aligned}$$

gives a contradiction. Therefore we conclude that no edge is visited more often than $\max\{1, 13\sqrt{t}/\delta\}$. Hence after t steps, at least $\min\{t, \sqrt{t}\delta/13\}$ distinct edges must be visited. Choosing $t := \max\{\mathcal{M}, 13^2 \mathcal{M}/\delta^2\}$, the deterministic walk visits at least \mathcal{M} distinct edges.

For the first claim, we observe that if the random walk visits at least \mathcal{M} distinct edges, then it also visits at least \mathcal{M}/Δ distinct vertices. Hence to visit \mathcal{N} vertices, we have to visit at least $\min\{\mathcal{N}\Delta, m\}$ edges. As shown above, to visit $\min\{\mathcal{N}\Delta, m\}$ edges, we have to spend

$$\min \{ \mathcal{O}(\mathcal{N}\Delta + (\mathcal{N}\Delta/\delta)^2), \mathcal{O}(m + (m/\delta)^2) \}$$

steps. □

At the end of Section 1.3, we give an example how the bounds for the random and deterministic walk compare to each other. For expander graphs, much stronger results are known. There, $\max_{u,v} H(u, v) = \mathcal{O}(n)$ and hence by a simple first-moment calculation one obtains that after $\mathcal{O}(n)$ steps, cn vertices are visited (where $0 < c < 1$ is any constant). We remark that a similar result can be shown for the deterministic walk, that is, after $\mathcal{O}(n \log \log n)$ steps it visits cn vertices of an expander graph (of constant degree).

6 Discussion

We have analyzed the vertex and edge cover time of the deterministic random walk and presented upper bounds for general graphs based on the local divergence, expansion properties, and flows. This is complemented with tight bounds for various common graph classes. It turns out that the deterministic random walk is surprisingly efficient though it has a strong adversary (as the order of the rotors is completely arbitrary) and it is not tailored for search problems (as it does not mark visited vertices).

In applications such as broadcasting [4, 26–28, 36] and sorting [9] the quasirandom version of the deterministic random walk seems to be especially efficient. There, the first rotor direction is chosen at random while the order of the rotors stays arbitrary. It would be interesting to quantify how much this quasirandom walk can cover all vertices or edges faster than our deterministic one.

References

- [1] D. Aldous. On the time taken by random walks on finite groups to visit every state. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, pages 361–374, 1983.
- [2] D. Aldous and J. Fill. *Reversible Markov Chains and Random Walks on Graphs*. (in preparation, draft available at <http://www.stat.berkeley.edu/aldous/RWG/book.html>), 2002.
- [3] R. Aleliunas, R. Karp, R. Lipton, L. Lovász, and C. Rackoff. Random walks, universal traversal sequences, and the complexity of maze problems. In *20th Annual IEEE Symposium on Foundations of Computer Science (FOCS '79)*, pages 218–223, 1979.
- [4] S. Angelopoulos, B. Doerr, A. Huber, and K. Panagiotou. Tight bounds for quasirandom rumor spreading. *The Electronic Journal of Combinatorics*, 16(1):R102, 2009.
- [5] C. Avin, M. Koucky, and Z. Lotker. How to explore a fast-changing world (cover time of a simple random walk on evolving graphs). In *35th International Colloquium on Automata, Languages, and Programming (ICALP '08)*, pages 121–132, 2008.

- [6] E. Bampas, L. Gasieniec, N. Hanusse, D. Ilcinkas, R. Klasing, and A. Kosowski. Euler tour lock-in problem in the rotor-router model. In *International Symposium on Distributed Computing (DISC'09)*, volume 5805 of *LNCS*, pages 423–435. Springer, 2009.
- [7] G. Barnes and U. Feige. Short random walks on graphs. In *25th Annual ACM Symposium on Theory of Computing (STOC '93)*, pages 728–737, 1993.
- [8] G. Barnes and U. Feige. Short random walks on graphs. *SIAM Journal of Discrete Mathematics*, 9(1):19–28, 1996.
- [9] R. D. Barve, E. F. Grove, and J. S. Vitter. Simple randomized mergesort on parallel disks. *Parallel Computing*, 23(4-5):601–631, 1997.
- [10] N. Biggs. *Algebraic Graph Theory*. Cambridge University Press, 1993.
- [11] A. Broder and A. Karlin. Bounds on the cover time. *Journal of Theoretical Probability*, 2(1):101–120, 1989.
- [12] E. Cesàro. Sur la multiplication des séries. *Bull. Sci. Math*, 14(2):114–120, 1890.
- [13] A. Chandra, P. Raghavan, W. Ruzzo, R. Smolensky, and P. Tiwari. The electrical resistance of a graph captures its commute and cover times. *Computational Complexity*, 6(4):312–340, 1997.
- [14] C. Cooper and A. Frieze. The cover time of random regular graphs. *SIAM Journal of Discrete Mathematics*, 18(4):728–740, 2005.
- [15] C. Cooper and A. Frieze. The cover time of two classes of random graphs. In *16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '05)*, pages 961–970, 2005.
- [16] C. Cooper and A. Frieze. The cover time of sparse random graphs. *Random Structures & Algorithms*, 30(1-2):1–16, 2007.
- [17] C. Cooper and A. Frieze. The cover time of the giant component of a random graph. *Random Structures & Algorithms*, 32(4):401–439, 2008.
- [18] C. Cooper and A. Frieze. The cover time of random geometric graphs. In *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '09)*, pages 48–57, 2009.
- [19] C. Cooper, D. Ilcinkas, R. Klasing, and A. Kosowski. Derandomizing random walks in undirected graphs using locally fair exploration strategies. In *36th International Colloquium on Automata, Languages, and Programming (ICALP '09)*, pages 411–422, 2009.
- [20] J. Cooper and J. Spencer. Simulating a random walk with constant error. *Combinatorics, Probability & Computing*, 15:815–822, 2006.
- [21] J. Cooper, B. Doerr, J. Spencer, and G. Tardos. Deterministic random walks on the integers. *European Journal of Combinatorics*, 28(8):2072–2090, 2007.
- [22] J. Cooper, B. Doerr, T. Friedrich, and J. Spencer. Deterministic random walks on regular trees. In *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '08)*, pages 766–772, 2008.

- [23] P. Diaconis and L. Saloff-Coste. Comparison theorems for reversible Markov chains. *Annals of Applied Probability*, 3(3):696–730, 1993.
- [24] P. Diaconis and D. Stroock. Geometric bounds for eigenvalues of markov chains. *Annals of Applied Probability*, 1(1):36–61, 1991.
- [25] B. Doerr and T. Friedrich. Deterministic random walks on the two-dimensional grid. *Combinatorics, Probability & Computing*, 18(1-2):123–144, 2009.
- [26] B. Doerr, T. Friedrich, and T. Sauerwald. Quasirandom rumor spreading. In *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '08)*, pages 773–781, 2008.
- [27] B. Doerr, T. Friedrich, M. Künnemann, and T. Sauerwald. Quasirandom rumor spreading: An experimental analysis. In *10th Workshop on Algorithm Engineering and Experiments (ALENEX '09)*, pages 145–153, 2009.
- [28] B. Doerr, T. Friedrich, and T. Sauerwald. Quasirandom rumor spreading: Expanders, push vs. pull, and robustness. In *36th International Colloquium on Automata, Languages, and Programming (ICALP '09)*, pages 366–377, 2009.
- [29] I. Dumitriu, P. Tetali, and P. Winkler. On playing golf with two balls. *SIAM J. Discrete Math.*, 16(4):604–615, 2003.
- [30] U. Feige. A tight lower bound for the cover time of random walks on graphs. *Random Structures & Algorithms*, 6(4):433–438, 1995.
- [31] U. Feige. Collecting coupons on trees, and the cover time of random walks. *Computational Complexity*, 6(4):341–356, 1997.
- [32] T. Friedrich and T. Sauerwald. Near-perfect load balancing by randomized rounding. In *41st Annual ACM Symposium on Theory of Computing (STOC '09)*, pages 121–130, 2009.
- [33] T. Friedrich, M. Gairing, and T. Sauerwald. Quasirandom load balancing. In *21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '10)*, pages 1620–1629, 2010.
- [34] L. Gasieniec, A. Pelc, T. Radzik, and X. Zhang. Tree exploration with logarithmic memory. In *18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '07)*, pages 585–594, 2007.
- [35] A. E. Holroyd and J. G. Propp. Rotor walks and Markov chains. *Algorithmic Probability and Combinatorics*, 2010. To appear. Also arXiv:0904.4507.
- [36] A. Huber and N. Fountoulakis. Quasirandom rumor spreading on the complete graph is as fast as randomized rumor spreading. *SIAM Journal on Discrete Mathematics*, 23(4):1964–1991, 2009.
- [37] J. Jonasson and O. Schramm. On the cover time of planar graphs. *Electronic Communications in Probability*, 5:85–90, 2000.
- [38] M. Kleber. Goldbug Variations. *The Mathematical Intelligencer*, 27, 2005.

- [39] R. E. Korf. Real-time heuristic search. *Artif. Intell.*, 42(2-3):189–211, 1990.
- [40] G. F. Lawler. Subdiffusive fluctuations for internal diffusion limited aggregation. *Ann. Probab.*, 23(1):71–86, 1995.
- [41] G. F. Lawler, M. Bramson, and D. Griffeath. Internal diffusion limited aggregation. *Ann. Probab.*, 20(4):2117–2140, 1992.
- [42] L. Levine and Y. Peres. Strong spherical asymptotics for rotor-router aggregation and the divisible sandpile. *Potential Analysis*, 30:1–27, 2009.
- [43] L. Lovász. Random walks on graphs: A survey. *Combinatorics, Paul Erdős is Eighty*, 2:1–46, 1993.
- [44] H. Meyerhenke and T. Sauerwald. Analyzing disturbed diffusion on networks. In *17th International Symposium on Algorithms and Computation (ISAAC '06)*, pages 429–438, 2006.
- [45] J. L. Palacios. Expected hitting and cover times of random walks on some special graphs. *Random Structures & Algorithms*, 5(1):173–182, 1994.
- [46] V. B. Priezzhev, D. Dhar, A. Dhar, and S. Krishnamurthy. Eulerian walkers as a model of self-organized criticality. *Phys. Rev. Lett.*, 77:5079–5082, 1996.
- [47] Y. Rabani, A. Sinclair, and R. Wanka. Local divergence of Markov chains and the analysis of iterative load balancing schemes. In *39th Annual IEEE Symposium on Foundations of Computer Science (FOCS '98)*, pages 694–705, 1998.
- [48] O. Reingold. Undirected connectivity in log-space. *J. ACM*, 55(4), 2008.
- [49] R. Rubinfeld. The cover time of a regular expander is $\mathcal{O}(n \log n)$. *Inf. Process. Lett.*, 35(1):49–51, 1990.
- [50] I. A. Wagner, M. Lindenbaum, and A. M. Bruckstein. Smell as a computational resource - a lesson we can learn from the ant. In *Israeli Symposium on Theory of Computing and Systems (ISTCS '96)*, pages 219–230, 1996.
- [51] I. A. Wagner, M. Lindenbaum, and A. M. Bruckstein. Distributed covering by ant-robots using evaporating traces. *IEEE Transactions on Robotics and Automation*, 15(5):918–933, 1999.
- [52] A. Wald. On cumulative sums of random variables. *Annals of Mathematical Statistics*, 15(3):283–296, 1944.
- [53] V. Yanovski, I. A. Wagner, and A. M. Bruckstein. A distributed ant algorithm for efficiently patrolling a network. *Algorithmica*, 37(3):165–186, 2003.
- [54] D. Zuckerman. On the time to traverse all edges of a graph. *Inf. Process. Lett.*, 38(6):335–337, 1991.
- [55] D. Zuckerman. A technique for lower bounding the cover time. *SIAM Journal on Discrete Mathematics*, 5(1):81–87, 1992.